MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
Appl. No. 10/738,407
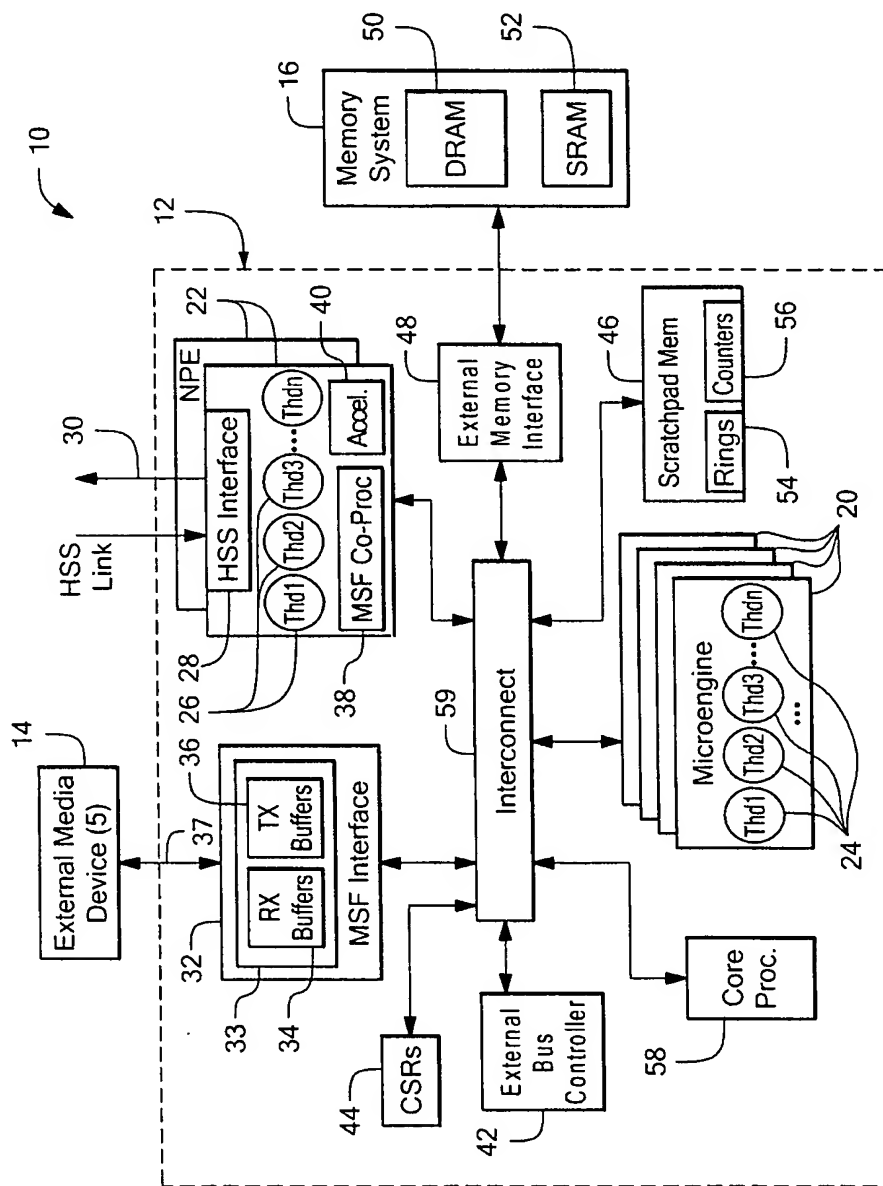Replacement Sheet

1/13

FIG. 1

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

2/13

FIG. 2

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

3/13

12, 64

| Software are Prepend Header | 70 |
| Data "Chunk" | |

72

FIG. 3A

| NPEID(8) | Byte count (8) | RES | V | E | S | C/F | Channel - ID (10) | 94 |
| RES | | | | | | | | 96 |

92   84   90   88   86   82   80

72

FIG. 3B

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
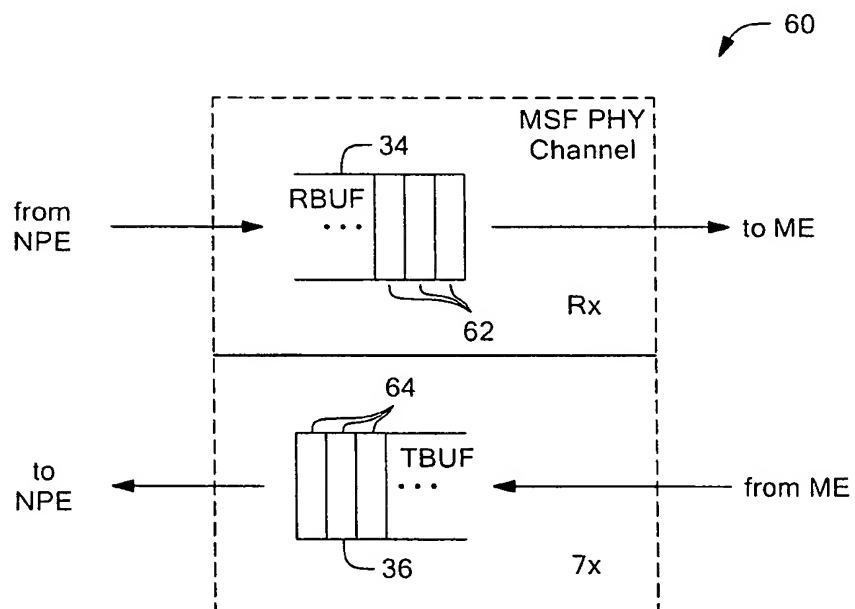Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

4/13

100

NPE receives data for a
channel from line-receive
hardware (e.g., HSS
coprocessor)
102

NPE processes and
reassembles frame-based
protocol data
104

End
of frame
received?
106
YES

NO

Programmed
# of bytes for this channel
reassembled?
108
NO

YES

NPE composes software
prepend header, setting all
fields correctly
110

NPE transmits prepend
header and frame data to
MSF RBUF element
112

ME Receive driver thread is
awakened by MSF, and
thread uses the prepend
header in the RBUF element
to process the frame data
(also in the RBUF element)
114

Following completion of
processing, ME receive
driver thread goes to sleep
to await another RBUF
element from MSF
116

FIG. 4

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
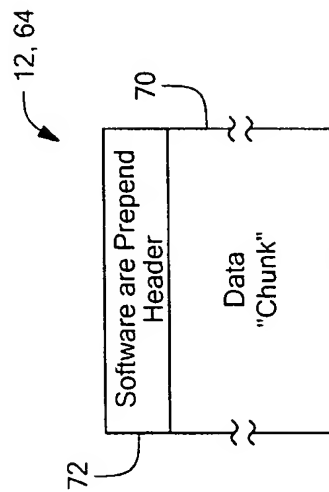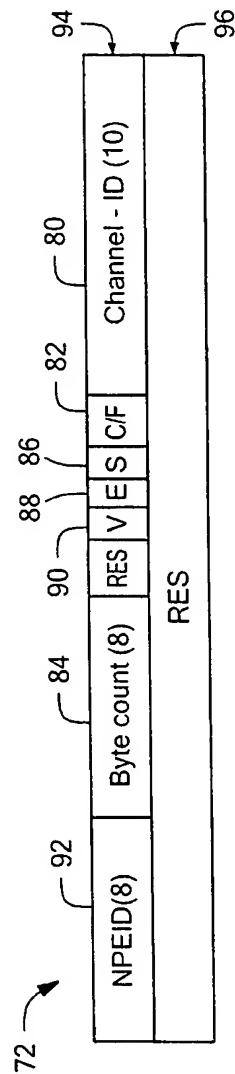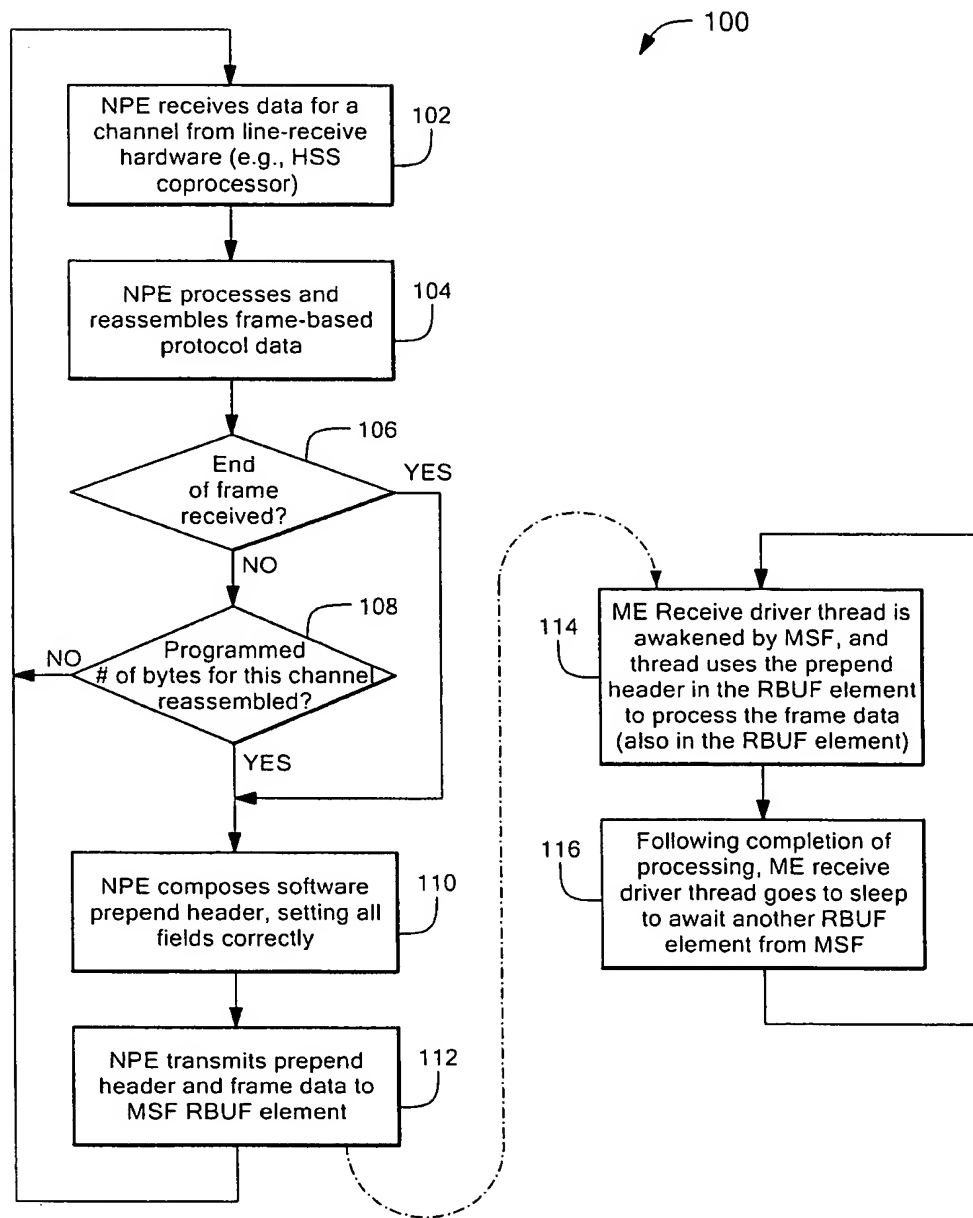Appl. No. 10/738,407
Replacement Sheet

5/13

```
                                                         120

      ┌──────────────────────┐
      │  NPE receives data for a │   122
      │  channel from line-receive │
      │  hardware (e.g., HSS     │
      │  coprocessor)            │
      └──────────────────────┘
                 │
      ┌──────────────────────┐
      │  NPE processes and       │   124
      │  reassembles cell-based  │
      │  protocol data           │
      └──────────────────────┘
                 │
              ╱──────╲   126
      NO    ╱  Entire   ╲
   ◄───────╱    cell      ╲
           ╲  received?  ╱
            ╲──────────╱
                 │ YES
      ┌──────────────────────┐
      │  NPE composes software   │   128
      │  prepend header, setting all │
      │  fields correctly        │
      └──────────────────────┘
                 │
      ┌──────────────────────┐
      │  NPE transmits prepend   │   130
      │  header and cell data to │
      │  MSF RBUF element        │
      └──────────────────────┘
```

```
              ┌─────────────────────────┐
      132     │  ME receive driver thread is │
              │  awakened by MSF, and       │
              │  thread uses the prepend    │
              │  header in the RBUF element │
              │  to process the cell data   │
              │  (also in the RBUF element) │
              └─────────────────────────┘
                          │
      134     ┌─────────────────────────┐
              │  Following completion of    │
              │  processing, ME receive     │
              │  driver thread goes to sleep │
              │  to await another RBUF      │
              │  element from MSF           │
              └─────────────────────────┘
```

FIG. 5

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
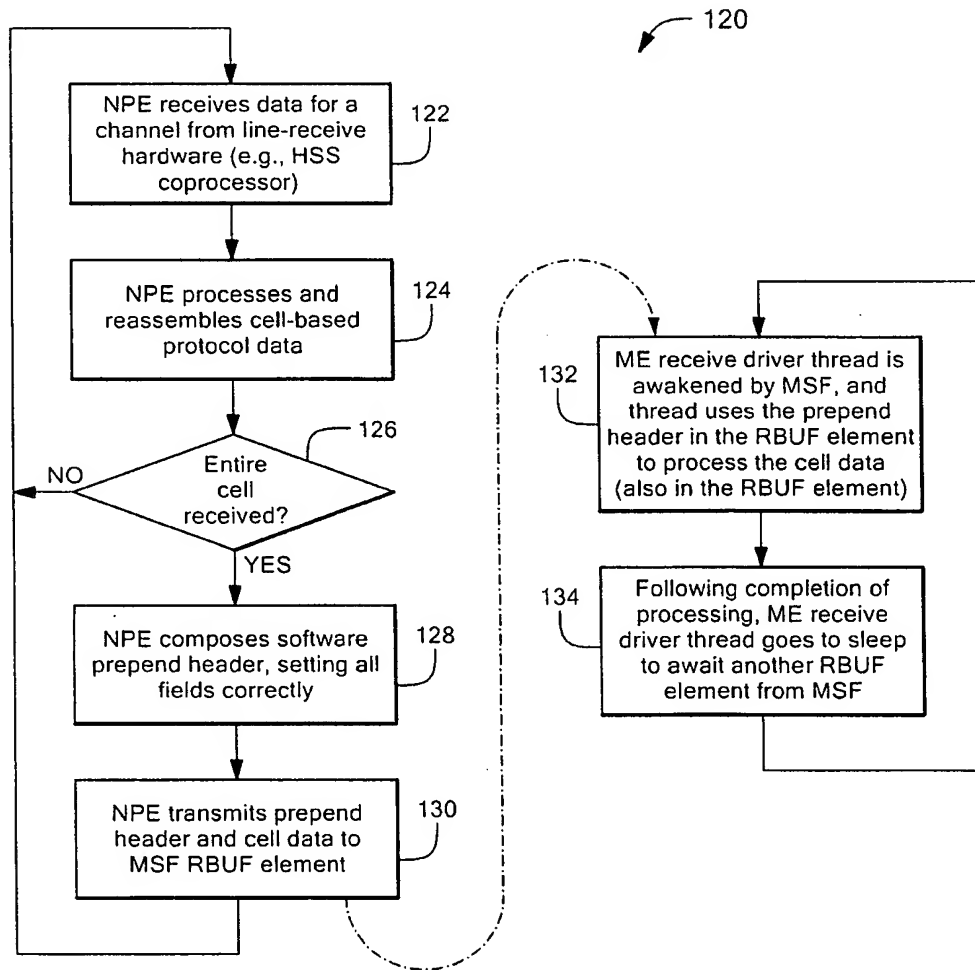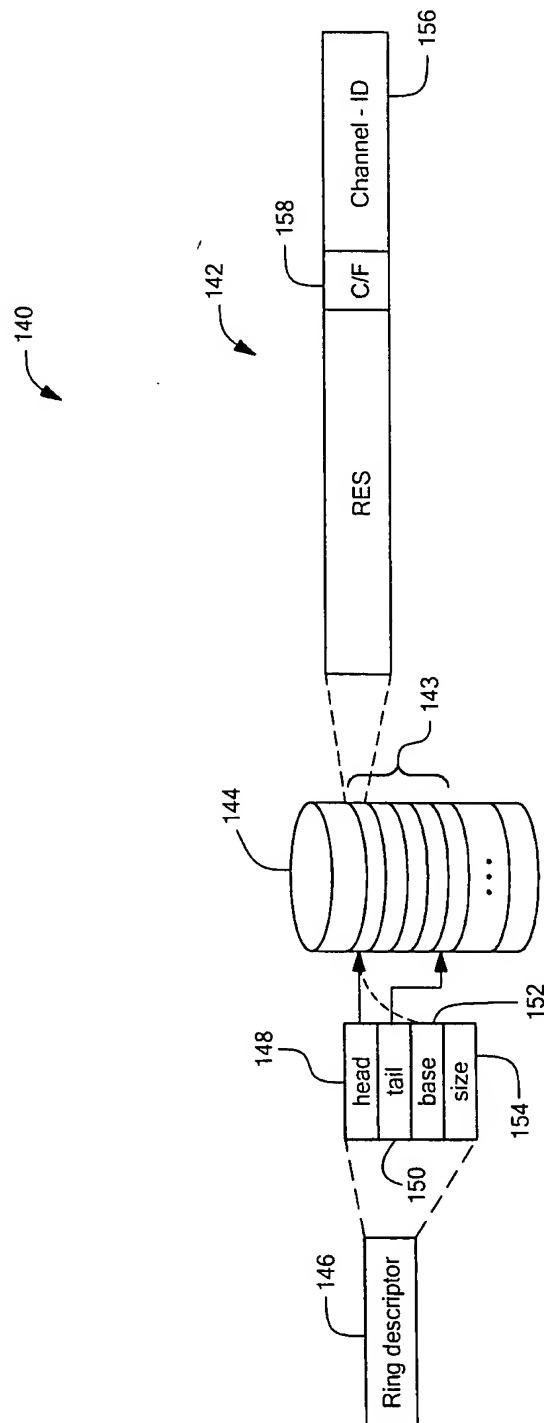Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

6/13

FIG. 6

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
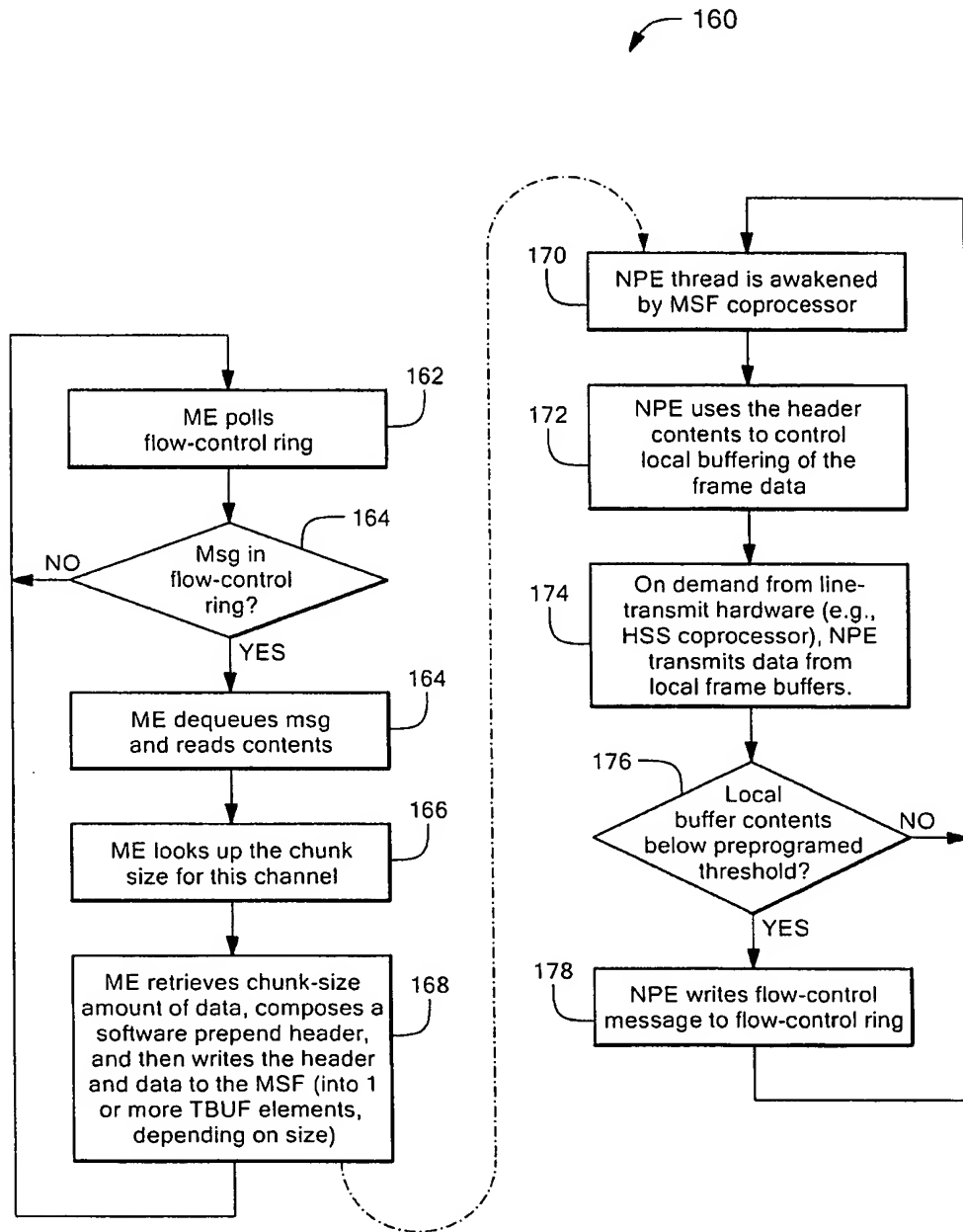Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

7/13

― 160

162
ME polls
flow-control ring

164
Msg in
flow-control
ring?

NO

YES

164
ME dequeues msg
and reads contents

166
ME looks up the chunk
size for this channel

168
ME retrieves chunk-size
amount of data, composes a
software prepend header,
and then writes the header
and data to the MSF (into 1
or more TBUF elements,
depending on size)

170
NPE thread is awakened
by MSF coprocessor

172
NPE uses the header
contents to control
local buffering of the
frame data

174
On demand from line-
transmit hardware (e.g.,
HSS coprocessor), NPE
transmits data from
local frame buffers.

176
Local
buffer contents
below preprogramed
threshold?

NO

YES

178
NPE writes flow-control
message to flow-control ring

FIG. 7

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
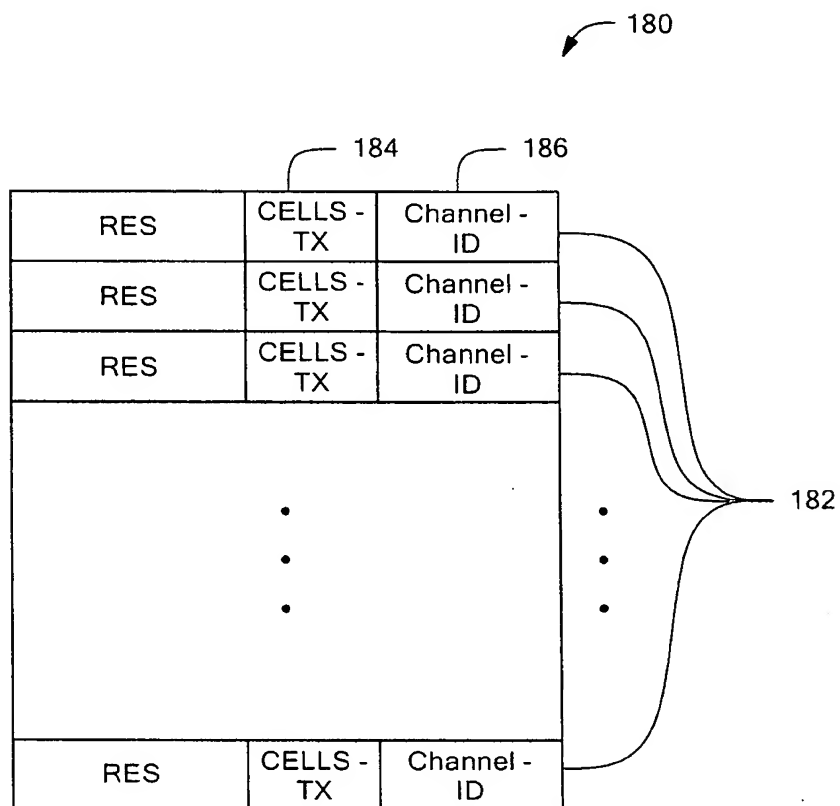Appl. No. 10/738,407
Replacement Sheet

8/13

RES	CELLS - TX	Channel - ID
RES	CELLS - TX	Channel - ID
RES	CELLS - TX	Channel - ID
RES	CELLS - TX	Channel - ID

FIG. 8

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
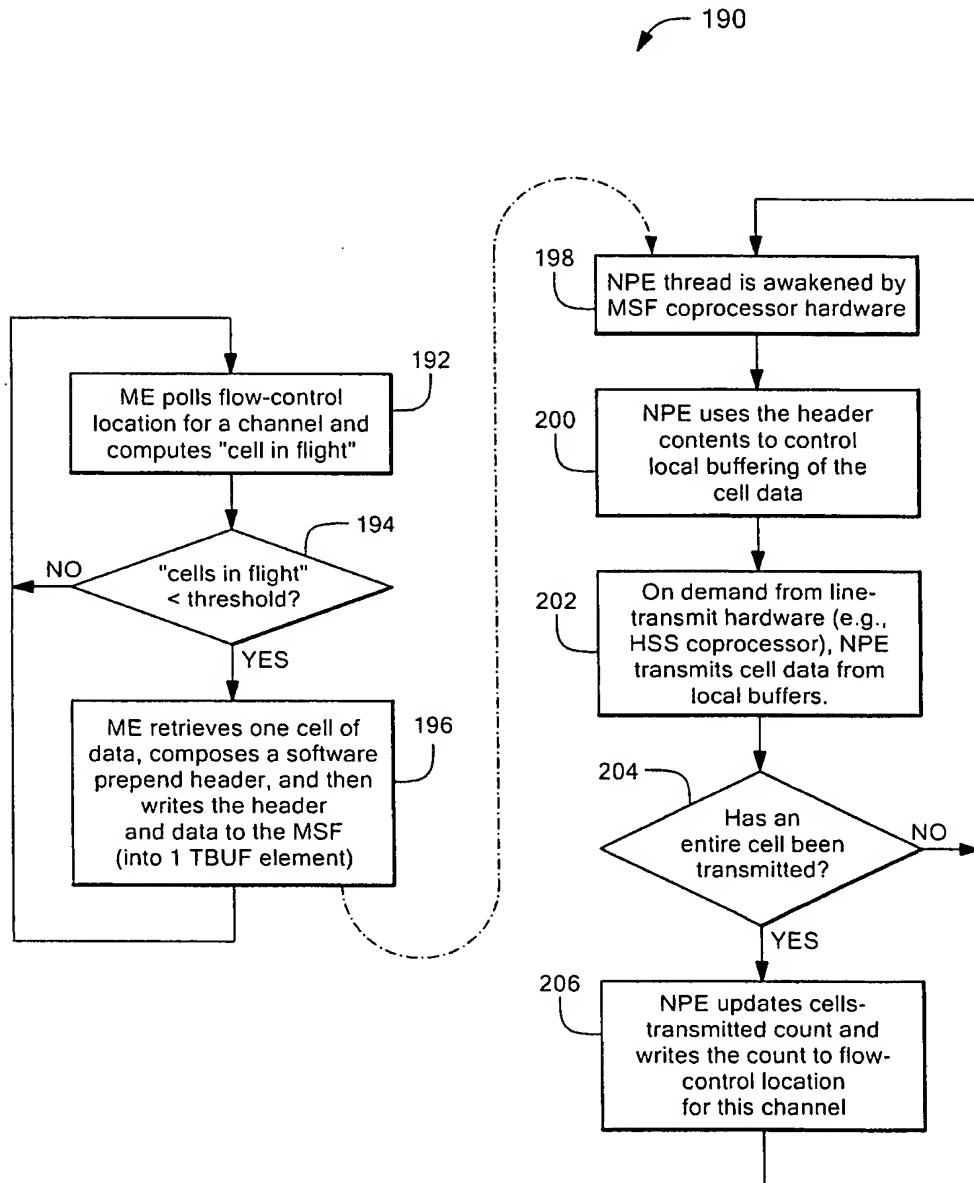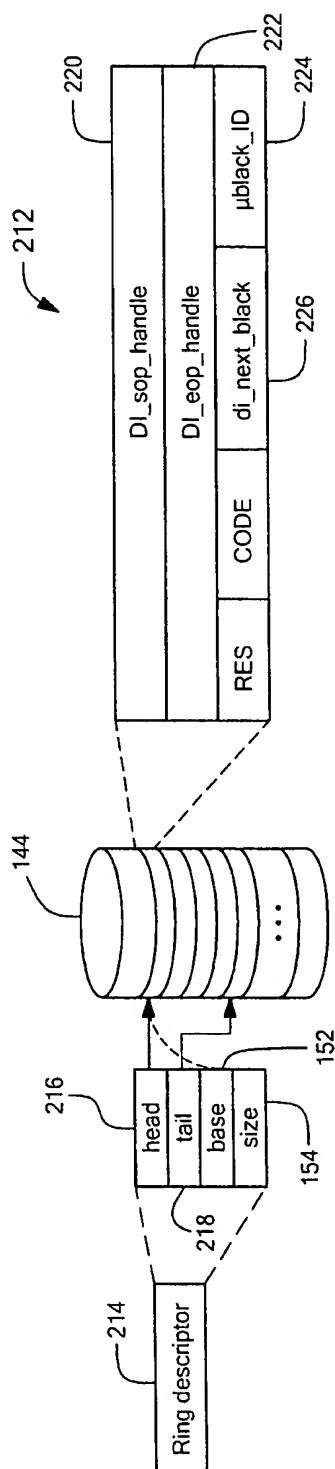Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

9/13

190

192 ME polls flow-control location for a channel and computes "cell in flight"

194 "cells in flight" < threshold? NO

YES

196 ME retrieves one cell of data, composes a software prepend header, and then writes the header and data to the MSF (into 1 TBUF element)

198 NPE thread is awakened by MSF coprocessor hardware

200 NPE uses the header contents to control local buffering of the cell data

202 On demand from line-transmit hardware (e.g., HSS coprocessor), NPE transmits cell data from local buffers.

204 Has an entire cell been transmitted? NO

YES

206 NPE updates cells-transmitted count and writes the count to flow-control location for this channel

FIG. 9

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

10/13

FIG. 10A

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
Muthu Venkatachalam et al.
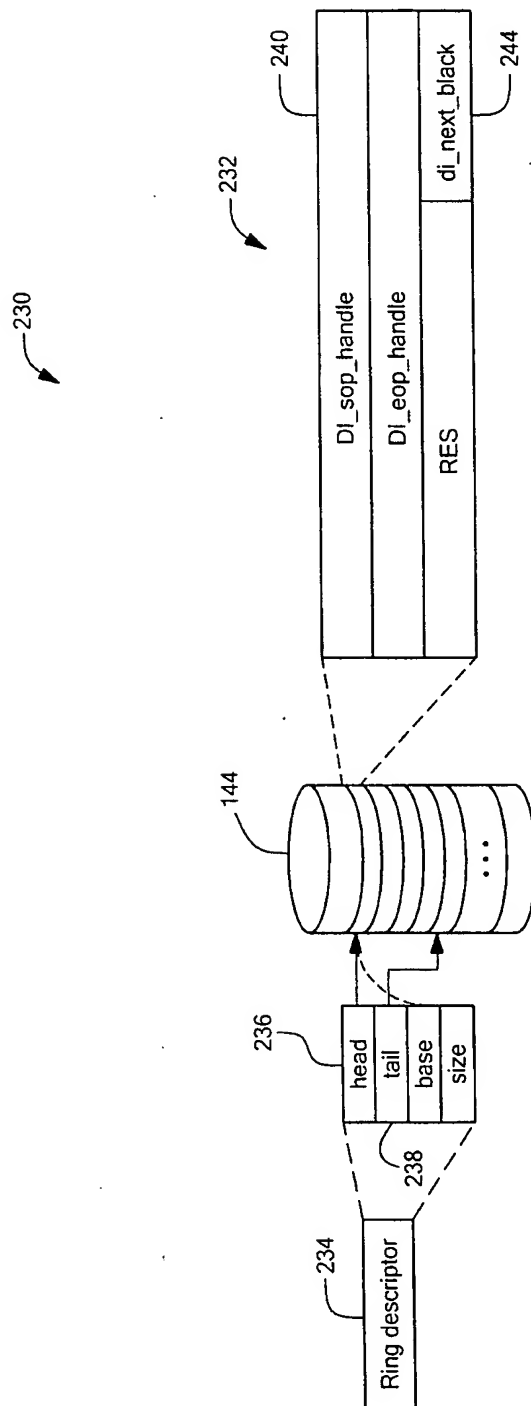Appl. No. 10/738,407
Replacement Sheet

11/13

FIG. 10B

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
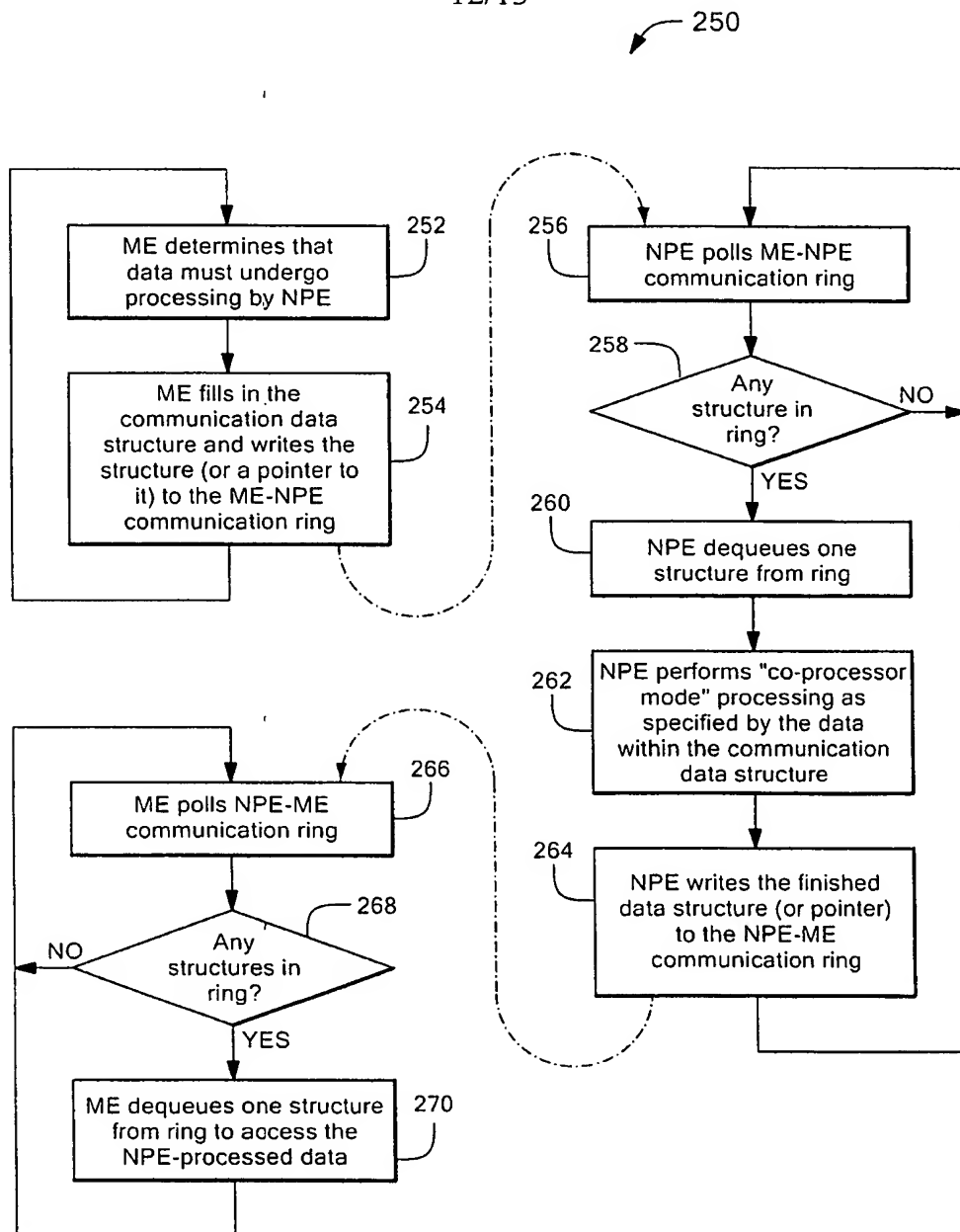Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

12/13

250

ME determines that
data must undergo
processing by NPE — 252

ME fills in the
communication data
structure and writes the
structure (or a pointer to
it) to the ME-NPE
communication ring — 254

256 — NPE polls ME-NPE
communication ring

258 — Any
structure in
ring? — NO

YES

260 — NPE dequeues one
structure from ring

262 — NPE performs "co-processor
mode" processing as
specified by the data
within the communication
data structure

264 — NPE writes the finished
data structure (or pointer)
to the NPE-ME
communication ring

266 — ME polls NPE-ME
communication ring

NO — 268 — Any
structures in
ring?

YES

ME dequeues one structure
from ring to access the
NPE-processed data — 270

FIG. 11

MICROENGINE TO NETWORK PROCESSING ENGINE INTERWORKING
FOR NETWORK PROCESSORS
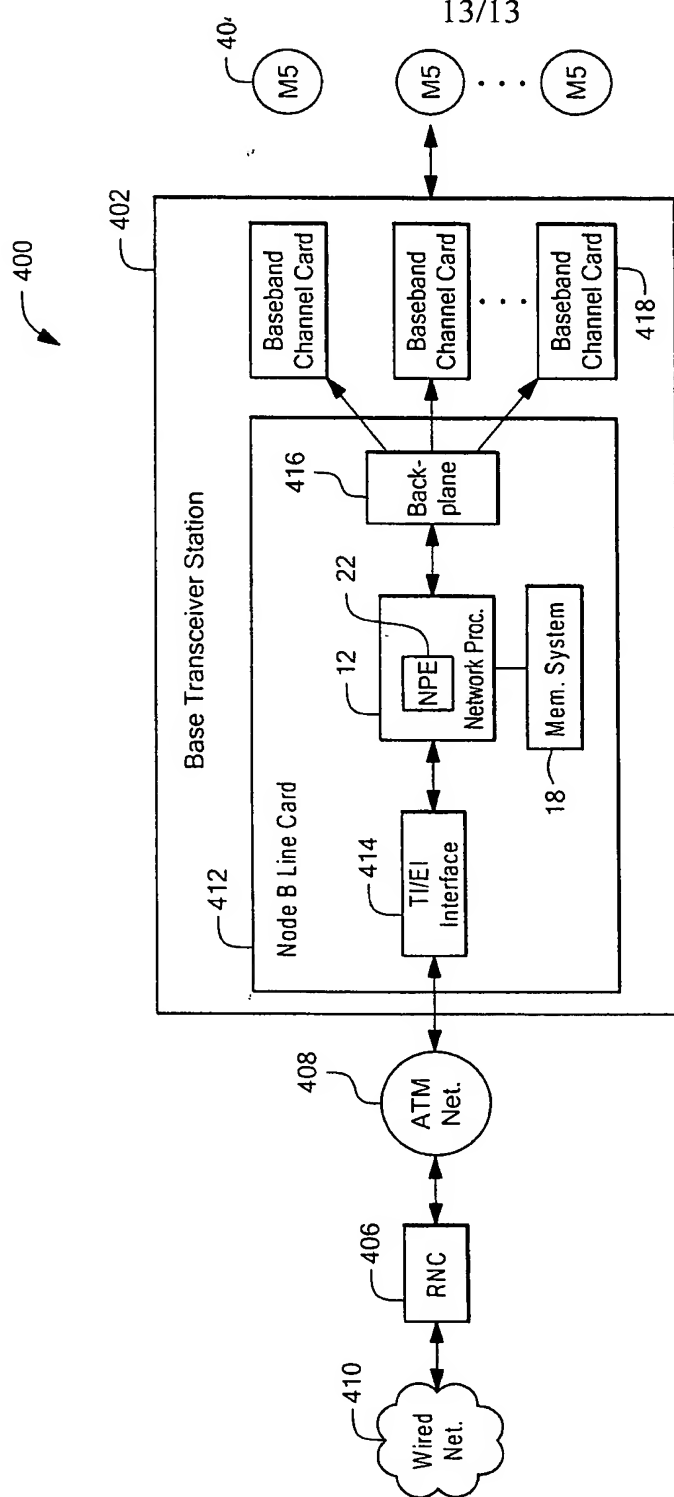Muthu Venkatachalam et al.
Appl. No. 10/738,407
Replacement Sheet

13/13

FIG. 12